Package: blorr (via r-universe)

September 7, 2024

Type Package

Title Tools for Developing Binary Logistic Regression Models

Version 0.3.0.9000

Description Tools designed to make it easier for beginner and intermediate users to build and validate binary logistic regression models. Includes bivariate analysis, comprehensive regression output, model fit statistics, variable selection procedures, model validation techniques and a 'shiny' app for interactive model building.

Depends R(>= 3.5)

Imports car, data.table, ggplot2, gridExtra, Rcpp, stats, utils

Suggests covr, grid, ineq, knitr, magrittr, rmarkdown, testthat, vdiffr, xplorerr

License MIT + file LICENSE

URL URL: https://blorr.rsquaredacademy.com/,

https://github.com/rsquaredacademy/blorr

BugReports https://github.com/rsquaredacademy/blorr/issues

VignetteBuilder knitr

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

LinkingTo Rcpp

Repository https://rsquaredacademy.r-universe.dev

RemoteUrl https://github.com/rsquaredacademy/blorr

RemoteRef HEAD

RemoteSha 073f672bb830080dd666c7cac4ff5d342b3ce0ac

Contents

`bank_marketing` *Bank marketing data set*

Description

The data is related with direct marketing campaigns of a Portuguese banking institution. The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required, in order to access if the product (bank term deposit) would be ('yes') or not ('no') subscribed.

Usage

```
bank_marketing
```

Format

A tibble with 4521 rows and 17 variables:

age age of the client

job type of job

marital marital status

education education level of the client

default has credit in default?

housing has housing loan?

loan has personal loan?

contact contact communication type

month last contact month of year

day_of_week last contact day of the week

duration last contact duration, in seconds

campaign number of contacts performed during this campaign and for this client

pdays number of days that passed by after the client was last contacted from a previous campaign

previous number of contacts performed before this campaign and for this clien

poutcome outcome of the previous marketing campaign

y has the client subscribed a term deposit?

Source

[Moro et al., 2014] S. Moro, P. Cortez and P. Rita. A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, Elsevier, 62:22-31, June 2014

`blorr` `blorr` *package*

Description

Tools for developing binary logistic regression models

Details

See the README on GitHub

`blr_bivariate_analysis` *Bivariate analysis*

Description

Information value and likelihood ratio chi square test for initial variable/predictor selection. Currently avialable for categorical predictors only.

Usage

```
blr_bivariate_analysis(data, response, ...)

## Default S3 method:
blr_bivariate_analysis(data, response, ...)
```

Arguments

`data`	A `tibble` or a `data.frame`.
`response`	Response variable; column in `data`.
`...`	Predictor variables; columns in `data`.

Value

A tibble with the following columns:

`Variable`	Variable name
`Information Value`	Information value
`LR Chi Square`	Likelihood ratio statisitc
`LR DF`	Likelihood ratio degrees of freedom
`LR p-value`	Likelihood ratio p value

See Also

Other bivariate analysis procedures: `blr_segment_dist()`, `blr_segment_twoway()`, `blr_segment()`, `blr_woe_iv_stats()`, `blr_woe_iv()`

Examples

```
blr_bivariate_analysis(hsb2, honcomp, female, prog, race, schtyp)
```

`blr_coll_diag` *Collinearity diagnostics*

Description

Variance inflation factor, tolerance, eigenvalues and condition indices.

Usage

```
blr_coll_diag(model)

blr_vif_tol(model)

blr_eigen_cindex(model)
```

Arguments

`model`	An object of class `glm`.

Details

Collinearity implies two variables are near perfect linear combinations of one another. Multicollinearity involves more than two variables. In the presence of multicollinearity, regression estimates are unstable and have high standard errors.

Tolerance

Percent of variance in the predictor that cannot be accounted for by other predictors.

Variance Inflation Factor

Variance inflation factors measure the inflation in the variances of the parameter estimates due to collinearities that exist among the predictors. It is a measure of how much the variance of the estimated regression coefficient β_k is inflated by the existence of correlation among the predictor variables in the model. A VIF of 1 means that there is no correlation among the kth predictor and the remaining predictor variables, and hence the variance of β_k is not inflated at all. The general rule of thumb is that VIFs exceeding 4 warrant further investigation, while VIFs exceeding 10 are signs of serious multicollinearity requiring correction.

Condition Index

Most multivariate statistical approaches involve decomposing a correlation matrix into linear combinations of variables. The linear combinations are chosen so that the first combination has the largest possible variance (subject to some restrictions), the second combination has the next largest variance, subject to being uncorrelated with the first, the third has the largest possible variance, subject to being uncorrelated with the first and second, and so forth. The variance of each of these linear combinations is called an eigenvalue. Collinearity is spotted by finding 2 or more variables that have large proportions of variance (.50 or more) that correspond to large condition indices. A rule of thumb is to label as large those condition indices in the range of 30 or larger.

Value

`blr_coll_diag` returns an object of class `"blr_coll_diag"`. An object of class `"blr_coll_diag"` is a list containing the following components:

`vif_t`	tolerance and variance inflation factors
`eig_cindex`	eigen values and condition index

References

Belsley, D. A., Kuh, E., and Welsch, R. E. (1980). Regression Diagnostics: Identifying Influential Data and Sources of Collinearity. New York: John Wiley & Sons.

Examples

```
# model
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

# vif and tolerance
blr_vif_tol(model)

# eigenvalues and condition indices
```

```
blr_eigen_cindex(model)

# collinearity diagnostics
blr_coll_diag(model)
```

blr_confusion_matrix *Confusion matrix*

Description

Confusion matrix and statistics.

Usage

```
blr_confusion_matrix(model, cutoff = 0.5, data = NULL, ...)

## Default S3 method:
blr_confusion_matrix(model, cutoff = 0.5, data = NULL, ...)
```

Arguments

model	An object of class glm.
cutoff	Cutoff for classification.
data	A tibble or a data.frame.
...	Other arguments.

Value

Confusion matix.

See Also

Other model validation techniques: blr_decile_capture_rate(), blr_decile_lift_chart(), blr_gains_table(), blr_gini_index(), blr_ks_chart(), blr_lorenz_curve(), blr_roc_curve(), blr_test_hosmer_lemeshow()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))

blr_confusion_matrix(model, cutoff = 0.4)
```

blr_decile_capture_rate
Event rate by decile

Description

Visualize the decile wise event rate.

Usage

```
blr_decile_capture_rate(
  gains_table,
  xaxis_title = "Decile",
  yaxis_title = "Capture Rate",
  title = "Capture Rate by Decile",
  bar_color = "blue",
  text_size = 3.5,
  text_vjust = -0.3,
  print_plot = TRUE
)
```

Arguments

gains_table	An object of class blr_gains_table.
xaxis_title	X axis title.
yaxis_title	Y axis title.
title	Plot title.
bar_color	Bar color.
text_size	Size of the bar labels.
text_vjust	Vertical justification of the bar labels.
print_plot	logical; if TRUE, prints the plot else returns a plot object.

See Also

Other model validation techniques: blr_confusion_matrix(), blr_decile_lift_chart(), blr_gains_table(), blr_gini_index(), blr_ks_chart(), blr_lorenz_curve(), blr_roc_curve(), blr_test_hosmer_lemeshow()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))
gt <- blr_gains_table(model)
blr_decile_capture_rate(gt)
```

blr_decile_lift_chart *Decile lift chart*

Description

Decile wise lift chart.

Usage

```
blr_decile_lift_chart(
  gains_table,
  xaxis_title = "Decile",
  yaxis_title = "Decile Mean / Global Mean",
  title = "Decile Lift Chart",
  bar_color = "blue",
  text_size = 3.5,
  text_vjust = -0.3,
  print_plot = TRUE
)
```

Arguments

gains_table	An object of class blr_gains_table.
xaxis_title	X axis title.
yaxis_title	Y axis title.
title	Plot title.
bar_color	Color of the bars.
text_size	Size of the bar labels.
text_vjust	Vertical justification of the bar labels.
print_plot	logical; if TRUE, prints the plot else returns a plot object.

See Also

Other model validation techniques: blr_confusion_matrix(), blr_decile_capture_rate(), blr_gains_table(), blr_gini_index(), blr_ks_chart(), blr_lorenz_curve(), blr_roc_curve(), blr_test_hosmer_lemeshow()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))
gt <- blr_gains_table(model)
blr_decile_lift_chart(gt)
```

`blr_gains_table` *Gains table & lift chart*

Description

Compute sensitivity, specificity, accuracy and KS statistics to generate the lift chart and the KS chart.

Usage

```
blr_gains_table(model, data = NULL)

## S3 method for class 'blr_gains_table'
plot(
  x,
  title = "Lift Chart",
  xaxis_title = "% Population",
  yaxis_title = "% Cumulative 1s",
  diag_line_col = "red",
  lift_curve_col = "blue",
  plot_title_justify = 0.5,
  print_plot = TRUE,
  ...
)
```

Arguments

`model`	An object of class `glm`.
`data`	A `tibble` or a `data.frame`.
`x`	An object of class `blr_gains_table`.
`title`	Plot title.
`xaxis_title`	X axis title.
`yaxis_title`	Y axis title.
`diag_line_col`	Diagonal line color.
`lift_curve_col`	Color of the lift curve.
`plot_title_justify`	Horizontal justification on the plot title.
`print_plot`	logical; if `TRUE`, prints the plot else returns a plot object.
`...`	Other inputs.

Value

A tibble.

References

Agresti, A. (2007), An Introduction to Categorical Data Analysis, Second Edition, New York: John Wiley & Sons.

Agresti, A. (2013), Categorical Data Analysis, Third Edition, New York: John Wiley & Sons.

Thomas LC (2009): Consumer Credit Models: Pricing, Profit, and Portfolio. Oxford, Oxford University Press.

Sobehart J, Keenan S, Stein R (2000): Benchmarking Quantitative Default Risk Models: A Validation Methodology, Moody's Investors Service.

See Also

Other model validation techniques: blr_confusion_matrix(), blr_decile_capture_rate(), blr_decile_lift_chart(), blr_gini_index(), blr_ks_chart(), blr_lorenz_curve(), blr_roc_curve(), blr_test_hosmer_lemeshow()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))
# gains table
blr_gains_table(model)

# lift chart
k <- blr_gains_table(model)
plot(k)
```

---

blr_gini_index *Gini index*

Description

Gini index is a measure of inequality and was developed to measure income inequality in labour market. In the predictive model, Gini Index is used for measuring discriminatory power.

Usage

```
blr_gini_index(model, data = NULL)
```

Arguments

model	An object of class glm.
data	A tibble or data.frame.

Value

Gini index.

References

Siddiqi N (2006): Credit Risk Scorecards: developing and implementing intelligent credit scoring. New Jersey, Wiley.

Müller M, Rönz B (2000): Credit Scoring using Semiparametric Methods. In: Franke J, Härdle W, Stahl G (Eds.): Measuring Risk in Complex Stochastic Systems. New York, Springer-Verlag.

https://doi.org/10.2753/REE1540-496X470605

See Also

Other model validation techniques: blr_confusion_matrix(), blr_decile_capture_rate(), blr_decile_lift_chart(), blr_gains_table(), blr_ks_chart(), blr_lorenz_curve(), blr_roc_curve(), blr_test_hosmer_lemeshow()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_gini_index(model)
```

---

blr_ks_chart *KS chart*

Description

Kolmogorov-Smirnov (KS) statistics is used to assess predictive power for marketing or credit risk models. It is the maximum difference between cumulative event and non-event distribution across score/probability bands. The gains table typically has across score bands and can be used to find the KS for a model.

Usage

```
blr_ks_chart(
  gains_table,
  title = "KS Chart",
  yaxis_title = " ",
  xaxis_title = "Cumulative Population %",
  ks_line_color = "black",
  print_plot = TRUE
)
```

Arguments

gains_table	An object of class blr_gains_table.
title	Plot title.
yaxis_title	Y axis title.
xaxis_title	X axis title.
ks_line_color	Color of the line indicating maximum KS statistic.
print_plot	logical; if TRUE, prints the plot else returns a plot object.

References

https://doi.org/10.1198/tast.2009.08210

https://www.ncbi.nlm.nih.gov/pubmed/843576

See Also

Other model validation techniques: blr_confusion_matrix(), blr_decile_capture_rate(), blr_decile_lift_chart(), blr_gains_table(), blr_gini_index(), blr_lorenz_curve(), blr_roc_curve(), blr_test_hosmer_lemeshow()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))
gt <- blr_gains_table(model)
blr_ks_chart(gt)
```

---

blr_launch_app *Launch shiny app*

Description

Launches shiny app for interactive model building.

Usage

```
blr_launch_app()
```

Examples

```
## Not run:
blr_launch_app()

## End(Not run)
```

blr_linktest *Model specification error*

Description

Test for model specification error.

Usage

```
blr_linktest(model)
```

Arguments

model	An object of class glm.

Value

An object of class glm.

References

Pregibon, D. 1979. Data analytic methods for generalized linear models. PhD diss., University of Toronto.

Pregibon, D. 1980. Goodness of link tests for generalized linear models.

Tukey, J. W. 1949. One degree of freedom for non-additivity.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_linktest(model)
```

blr_lorenz_curve *Lorenz curve*

Description

Lorenz curve is a visual representation of inequality. It is used to measure the discriminatory power of the predictive model.

Usage

```
blr_lorenz_curve(
  model,
  data = NULL,
  title = "Lorenz Curve",
  xaxis_title = "Cumulative Events %",
  yaxis_title = "Cumulative Non Events %",
  diag_line_col = "red",
  lorenz_curve_col = "blue",
  print_plot = TRUE
)
```

Arguments

model	An object of class glm.
data	A tibble or data.frame.
title	Plot title.
xaxis_title	X axis title.
yaxis_title	Y axis title.
diag_line_col	Diagonal line color.
lorenz_curve_col	Color of the lorenz curve.
print_plot	logical; if TRUE, prints the plot else returns a plot object.

See Also

Other model validation techniques: blr_confusion_matrix(), blr_decile_capture_rate(), blr_decile_lift_chart(), blr_gains_table(), blr_gini_index(), blr_ks_chart(), blr_roc_curve(), blr_test_hosmer_lemeshow()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_lorenz_curve(model)
```

---

blr_model_fit_stats *Model fit statistics*

Description

Model fit statistics.

Usage

```
blr_model_fit_stats(model, ...)
```

Arguments

model	An object of class glm.
...	Other inputs.

References

Menard, S. (2000). Coefficients of determination for multiple logistic regression analysis. The American Statistician, 54(1), 17-24.

Windmeijer, F. A. G. (1995). Goodness-of-fit measures in binary choice models. Econometric Reviews, 14, 101-116.

Hosmer, D.W., Jr., & Lemeshow, S. (2000), Applied logistic regression(2nd ed.). New York: John Wiley & Sons.

J. Scott Long & Jeremy Freese, 2000. "FITSTAT: Stata module to compute fit statistics for single equation regression models," Statistical Software Components S407201, Boston College Department of Economics, revised 22 Feb 2001.

Freese, Jeremy and J. Scott Long. Regression Models for Categorical Dependent Variables Using Stata. College Station: Stata Press, 2006.

Long, J. Scott. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks: Sage Publications, 1997.

See Also

Other model fit statistics: blr_multi_model_fit_stats(), blr_pairs(), blr_rsq_adj_count(), blr_rsq_cox_snell(), blr_rsq_effron(), blr_rsq_mcfadden_adj(), blr_rsq_mckelvey_zavoina(), blr_rsq_nagelkerke(), blr_test_lr()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))

blr_model_fit_stats(model)
```

---

blr_multi_model_fit_stats

Multi model fit statistics

Description

Measures of model fit statistics for multiple models.

Usage

```
blr_multi_model_fit_stats(model, ...)

## Default S3 method:
blr_multi_model_fit_stats(model, ...)
```

Arguments

model	An object of class glm.
...	Objects of class glm.

Value

A tibble.

See Also

Other model fit statistics: blr_model_fit_stats(), blr_pairs(), blr_rsq_adj_count(), blr_rsq_cox_snell(), blr_rsq_effron(), blr_rsq_mcfadden_adj(), blr_rsq_mckelvey_zavoina(), blr_rsq_nagelkerke(), blr_test_lr()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

model2 <- glm(honcomp ~ female + read + math, data = hsb2,
family = binomial(link = 'logit'))

blr_multi_model_fit_stats(model, model2)
```

---

blr_pairs *Concordant & discordant pairs*

Description

Association of predicted probabilities and observed responses.

Usage

```
blr_pairs(model)
```

Arguments

model	An object of class glm.

Value

A tibble.

References

https://doi.org/10.1080/10485259808832744

https://doi.org/10.1177/1536867X0600600302

See Also

Other model fit statistics: blr_model_fit_stats(), blr_multi_model_fit_stats(), blr_rsq_adj_count(), blr_rsq_cox_snell(), blr_rsq_effron(), blr_rsq_mcfadden_adj(), blr_rsq_mckelvey_zavoina(), blr_rsq_nagelkerke(), blr_test_lr()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_pairs(model)
```

---

blr_plot_c_fitted *CI Displacement C vs fitted values plot*

Description

Confidence interval displacement diagnostics C vs fitted values plot.

Usage

```
blr_plot_c_fitted(
  model,
  point_color = "blue",
  title = "CI Displacement C vs Fitted Values Plot",
  xaxis_title = "Fitted Values",
  yaxis_title = "CI Displacement C"
)
```

Arguments

model	An object of class glm.
point_color	Color of the points.
title	Title of the plot.
xaxis_title	X axis label.
yaxis_title	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_c_fitted(model)
```

---

blr_plot_c_leverage   *CI Displacement C vs leverage plot*

Description

Confidence interval displacement diagnostics C vs leverage plot.

Usage

```
blr_plot_c_leverage(
  model,
  point_color = "blue",
  title = "CI Displacement C vs Leverage Plot",
  xaxis_title = "Leverage",
  yaxis_title = "CI Displacement C"
)
```

Arguments

model	An object of class glm.
point_color	Color of the points.
title	Title of the plot.
xaxis_title	X axis label.
yaxis_title	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_c_leverage(model)
```

---

blr_plot_deviance_fitted
Deviance vs fitted values plot

Description

Deviance vs fitted values plot.

Usage

```
blr_plot_deviance_fitted(
  model,
  point_color = "blue",
  line_color = "red",
  title = "Deviance Residual vs Fitted Values",
  xaxis_title = "Fitted Values",
  yaxis_title = "Deviance Residual"
)
```

Arguments

`model`	An object of class `glm`.
`point_color`	Color of the points.
`line_color`	Color of the horizontal line.
`title`	Title of the plot.
`xaxis_title`	X axis label.
`yaxis_title`	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_deviance_fitted(model)
```

---

blr_plot_deviance_residual
Deviance residual values

Description

Deviance residuals plot.

Usage

```
blr_plot_deviance_residual(
  model,
  point_color = "blue",
  title = "Deviance Residuals Plot",
  xaxis_title = "id",
  yaxis_title = "Deviance Residuals"
)
```

Arguments

model	An object of class glm.
point_color	Color of the points.
title	Title of the plot.
xaxis_title	X axis label.
yaxis_title	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_deviance_residual(model)
```

---

blr_plot_dfbetas_panel

DFBETAs panel

Description

Panel of plots to detect influential observations using DFBETAs.

Usage

```
blr_plot_dfbetas_panel(model, print_plot = TRUE)
```

Arguments

model	An object of class glm.
print_plot	logical; if TRUE, prints the plot else returns a plot object.

Details

DFBETA measures the difference in each parameter estimate with and without the influential point. There is a DFBETA for each data point i.e if there are n observations and k variables, there will be $n * k$ DFBETAs. In general, large values of DFBETAS indicate observations that are influential in estimating a given parameter. Belsley, Kuh, and Welsch recommend 2 as a general cutoff value to indicate influential observations and $2/\sqrt(n)$ as a size-adjusted cutoff.

Value

list; `blr_dfbetas_panel` returns a list of tibbles (for intercept and each predictor) with the observation number and DFBETA of observations that exceed the threshold for classifying an observation as an outlier/influential observation.

References

Belsley, David A.; Kuh, Edwin; Welsh, Roy E. (1980). Regression Diagnostics: Identifying Influential Data and Sources of Collinearity. Wiley Series in Probability and Mathematical Statistics. New York: John Wiley & Sons. pp. ISBN 0-471-05856-4.

Examples

```
## Not run:
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_dfbetas_panel(model)

## End(Not run)
```

`blr_plot_diag_c` *CI Displacement C plot*

Description

Confidence interval displacement diagnostics C plot.

Usage

```
blr_plot_diag_c(
  model,
  point_color = "blue",
  title = "CI Displacement C Plot",
  xaxis_title = "id",
  yaxis_title = "CI Displacement C"
)
```

Arguments

`model`	An object of class `glm`.
`point_color`	Color of the points.
`title`	Title of the plot.
`xaxis_title`	X axis label.
`yaxis_title`	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_diag_c(model)
```

`blr_plot_diag_cbar` *CI Displacement CBAR plot*

Description

Confidence interval displacement diagnostics CBAR plot.

Usage

```
blr_plot_diag_cbar(
  model,
  point_color = "blue",
  title = "CI Displacement CBAR Plot",
  xaxis_title = "id",
  yaxis_title = "CI Displacement CBAR"
)
```

Arguments

`model`	An object of class `glm`.
`point_color`	Color of the points.
`title`	Title of the plot.
`xaxis_title`	X axis label.
`yaxis_title`	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_diag_cbar(model)
```

blr_plot_diag_difchisq *Delta chisquare plot*

Description

Diagnostics for detecting ill fitted observations.

Usage

```
blr_plot_diag_difchisq(
  model,
  point_color = "blue",
  title = "Delta Chisquare Plot",
  xaxis_title = "id",
  yaxis_title = "Delta Chisquare"
)
```

Arguments

model	An object of class glm.
point_color	Color of the points.
title	Title of the plot.
xaxis_title	X axis label.
yaxis_title	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_diag_difchisq(model)
```

blr_plot_diag_difdev *Delta deviance plot*

Description

Diagnostics for detecting ill fitted observations.

Usage

```
blr_plot_diag_difdev(
  model,
  point_color = "blue",
  title = "Delta Deviance Plot",
  xaxis_title = "id",
  yaxis_title = "Delta Deviance"
)
```

Arguments

`model`	An object of class `glm`.
`point_color`	Color of the points.
`title`	Title of the plot.
`xaxis_title`	X axis label.
`yaxis_title`	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_diag_difdev(model)
```

`blr_plot_diag_fit` *Fitted values diagnostics plot*

Description

Diagnostic plots for fitted values.

Usage

```
blr_plot_diag_fit(model, print_plot = TRUE)
```

Arguments

`model`	An object of class `glm`.
`print_plot`	logical; if `TRUE`, prints the plot else returns a plot object.

Value

A panel of diagnostic plots for fitted values.

References

Fox, John (1991), Regression Diagnostics. Newbury Park, CA: Sage Publications.

Cook, R. D. and Weisberg, S. (1982), Residuals and Influence in Regression, New York: Chapman & Hall.

See Also

Other diagnostic plots: blr_plot_diag_influence(), blr_plot_diag_leverage()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_diag_fit(model)
```

---

blr_plot_diag_influence

Influence diagnostics plot

Description

Reisudal diagnostic plots for detecting influential observations.

Usage

```
blr_plot_diag_influence(model, print_plot = TRUE)
```

Arguments

model	An object of class glm.
print_plot	logical; if TRUE, prints the plot else returns a plot object.

Value

A panel of influence diagnostic plots.

References

Fox, John (1991), Regression Diagnostics. Newbury Park, CA: Sage Publications.

Cook, R. D. and Weisberg, S. (1982), Residuals and Influence in Regression, New York: Chapman & Hall.

See Also

Other diagnostic plots: blr_plot_diag_fit(), blr_plot_diag_leverage()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_diag_influence(model)
```

`blr_plot_diag_leverage` *Leverage diagnostics plot*

Description

Diagnostic plots for leverage.

Usage

```
blr_plot_diag_leverage(model, print_plot = TRUE)
```

Arguments

model	An object of class glm.
print_plot	logical; if TRUE, prints the plot else returns a plot object.

Value

A panel of diagnostic plots for leverage.

References

Fox, John (1991), Regression Diagnostics. Newbury Park, CA: Sage Publications.

Cook, R. D. and Weisberg, S. (1982), Residuals and Influence in Regression, New York: Chapman & Hall.

See Also

Other diagnostic plots: blr_plot_diag_fit(), blr_plot_diag_influence()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_diag_leverage(model)
```

blr_plot_difchisq_fitted

Delta chi square vs fitted values plot

Description

Delta Chi Square vs fitted values plot for detecting ill fitted observations.

Usage

```
blr_plot_difchisq_fitted(
  model,
  point_color = "blue",
  title = "Delta Chi Square vs Fitted Values Plot",
  xaxis_title = "Fitted Values",
  yaxis_title = "Delta Chi Square"
)
```

Arguments

model	An object of class glm.
point_color	Color of the points.
title	Title of the plot.
xaxis_title	X axis label.
yaxis_title	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_difchisq_fitted(model)
```

blr_plot_difchisq_leverage

Delta chi square vs leverage plot

Description

Delta chi square vs leverage plot.

Usage

```
blr_plot_difchisq_leverage(
  model,
  point_color = "blue",
  title = "Delta Chi Square vs Leverage Plot",
  xaxis_title = "Leverage",
  yaxis_title = "Delta Chi Square"
)
```

Arguments

model	An object of class glm.
point_color	Color of the points.
title	Title of the plot.
xaxis_title	X axis label.
yaxis_title	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_difchisq_leverage(model)
```

---

blr_plot_difdev_fitted

Delta deviance vs fitted values plot

Description

Delta deviance vs fitted values plot for detecting ill fitted observations.

Usage

```
blr_plot_difdev_fitted(
  model,
  point_color = "blue",
  title = "Delta Deviance vs Fitted Values Plot",
  xaxis_title = "Fitted Values",
  yaxis_title = "Delta Deviance"
)
```

Arguments

model	An object of class glm.
point_color	Color of the points.
title	Title of the plot.
xaxis_title	X axis label.
yaxis_title	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_difdev_fitted(model)
```

---

blr_plot_difdev_leverage

Delta deviance vs leverage plot

Description

Delta deviance vs leverage plot.

Usage

```
blr_plot_difdev_leverage(
  model,
  point_color = "blue",
  title = "Delta Deviance vs Leverage Plot",
  xaxis_title = "Leverage",
  yaxis_title = "Delta Deviance"
)
```

Arguments

model	An object of class glm.
point_color	Color of the points.
title	Title of the plot.
xaxis_title	X axis label.
yaxis_title	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_difdev_leverage(model)
```

`blr_plot_fitted_leverage`
Fitted values vs leverage plot

Description

Fitted values vs leverage plot.

Usage

```
blr_plot_fitted_leverage(
  model,
  point_color = "blue",
  title = "Fitted Values vs Leverage Plot",
  xaxis_title = "Leverage",
  yaxis_title = "Fitted Values"
)
```

Arguments

`model`	An object of class `glm`.
`point_color`	Color of the points.
`title`	Title of the plot.
`xaxis_title`	X axis label.
`yaxis_title`	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_fitted_leverage(model)
```

blr_plot_leverage *Leverage plot*

Description

Leverage plot.

Usage

```
blr_plot_leverage(
  model,
  point_color = "blue",
  title = "Leverage Plot",
  xaxis_title = "id",
  yaxis_title = "Leverage"
)
```

Arguments

model	An object of class glm.
point_color	Color of the points.
title	Title of the plot.
xaxis_title	X axis label.
yaxis_title	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_leverage(model)
```

blr_plot_leverage_fitted *Leverage vs fitted values plot*

Description

Leverage vs fitted values plot

Usage

```
blr_plot_leverage_fitted(
  model,
  point_color = "blue",
  title = "Leverage vs Fitted Values",
  xaxis_title = "Fitted Values",
  yaxis_title = "Leverage"
)
```

Arguments

`model`	An object of class `glm`.
`point_color`	Color of the points.
`title`	Title of the plot.
`xaxis_title`	X axis label.
`yaxis_title`	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_leverage_fitted(model)
```

`blr_plot_pearson_residual`
Residual values plot

Description

Standardised pearson residuals plot.

Usage

```
blr_plot_pearson_residual(
  model,
  point_color = "blue",
  title = "Standardized Pearson Residuals",
  xaxis_title = "id",
  yaxis_title = "Standardized Pearson Residuals"
)
```

Arguments

model	An object of class glm.
point_color	Color of the points.
title	Title of the plot.
xaxis_title	X axis label.
yaxis_title	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_pearson_residual(model)
```

---

blr_plot_residual_fitted

Residual vs fitted values plot

Description

Residual vs fitted values plot.

Usage

```
blr_plot_residual_fitted(
  model,
  point_color = "blue",
  line_color = "red",
  title = "Standardized Pearson Residual vs Fitted Values",
  xaxis_title = "Fitted Values",
  yaxis_title = "Standardized Pearson Residual"
)
```

Arguments

model	An object of class glm.
point_color	Color of the points.
line_color	Color of the horizontal line.
title	Title of the plot.
xaxis_title	X axis label.
yaxis_title	Y axis label.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_plot_residual_fitted(model)
```

---

blr_prep_dcrate_data *Decile capture rate data*

Description

Data for generating decile capture rate.

Usage

```
blr_prep_dcrate_data(gains_table)
```

Arguments

gains_table	An object of clas blr_gains_table

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))
gt <- blr_gains_table(model)
blr_prep_dcrate_data(gt)
```

---

blr_prep_kschart_data *KS Chart data*

Description

Data for generating KS chart.

Usage

```
blr_prep_kschart_data(gains_table)

blr_prep_kschart_line(gains_table)

blr_prep_ksannotate_y(ks_line)

blr_prep_kschart_stat(ks_line)

blr_prep_ksannotate_x(ks_line)
```

Arguments

gains_table	An object of clas blr_gains_table.
ks_line	Overall conversion rate.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))
gt <- blr_gains_table(model)
blr_prep_kschart_data(gt)
ks_line <- blr_prep_kschart_line(gt)
blr_prep_kschart_stat(ks_line)
blr_prep_ksannotate_y(ks_line)
blr_prep_ksannotate_x(ks_line)
```

---

blr_prep_lchart_gmean *Lift Chart data*

Description

Data for generating lift chart.

Usage

```
blr_prep_lchart_gmean(gains_table)

blr_prep_lchart_data(gains_table, global_mean)
```

Arguments

gains_table	An object of clas blr_gains_table.
global_mean	Overall conversion rate.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))
gt <- blr_gains_table(model)
globalmean <- blr_prep_lchart_gmean(gt)
blr_prep_lchart_data(gt, globalmean)
```

blr_prep_lorenz_data *Lorenz curve data*

Description

Data for generating Lorenz curve.

Usage

```
blr_prep_lorenz_data(model, data = NULL, test_data = FALSE)
```

Arguments

model	An object of class glm.
data	A tibble or data.frame.
test_data	Logical; TRUE if data is test data and FALSE if training data.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))
data <- model$data
blr_prep_lorenz_data(model, data, FALSE)
```

blr_prep_roc_data *ROC curve data*

Description

Data for generating ROC curve.

Usage

```
blr_prep_roc_data(gains_table)
```

Arguments

gains_table	An object of clas blr_gains_table

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))
gt <- blr_gains_table(model)
blr_prep_roc_data(gt)
```

blr_regress *Binary logistic regression*

Description

Binary logistic regression.

Usage

```
blr_regress(object, ...)

## S3 method for class 'glm'
blr_regress(object, odd_conf_limit = FALSE, ...)
```

Arguments

object	An object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted or class glm.
...	Other inputs.
odd_conf_limit	If TRUE, odds ratio confidence limts will be displayed.

Examples

```
# using formula
blr_regress(object = honcomp ~ female + read + science, data = hsb2)

# using a model built with glm
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))

blr_regress(model)

# odds ratio estimates
blr_regress(model, odd_conf_limit = TRUE)
```

blr_residual_diagnostics *Residual diagnostics*

Description

Diagnostics for confidence interval displacement and detecting ill fitted observations.

Usage

```
blr_residual_diagnostics(model)
```

Arguments

`model` An object of class `glm`.

Value

C, CBAR, DIFDEV and DIFCHISQ.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

blr_residual_diagnostics(model)
```

---

blr_roc_curve *ROC curve*

Description

Receiver operating characteristic curve (ROC) curve is used for assessing accuracy of the model classification.

Usage

```
blr_roc_curve(
  gains_table,
  title = "ROC Curve",
  xaxis_title = "1 - Specificity",
  yaxis_title = "Sensitivity",
  roc_curve_col = "blue",
  diag_line_col = "red",
  point_shape = 18,
  point_fill = "blue",
  point_color = "blue",
  plot_title_justify = 0.5,
  print_plot = TRUE
)
```

Arguments

`gains_table`	An object of class `blr_gains_table`.
`title`	Plot title.
`xaxis_title`	X axis title.
`yaxis_title`	Y axis title.
`roc_curve_col`	Color of the roc curve.

`diag_line_col`	Diagonal line color.
`point_shape`	Shape of the points on the roc curve.
`point_fill`	Fill of the points on the roc curve.
`point_color`	Color of the points on the roc curve.
`plot_title_justify`	Horizontal justification on the plot title.
`print_plot`	logical; if `TRUE`, prints the plot else returns a plot object.

References

Agresti, A. (2007), An Introduction to Categorical Data Analysis, Second Edition, New York: John Wiley & Sons.

Hosmer, D. W., Jr. and Lemeshow, S. (2000), Applied Logistic Regression, 2nd Edition, New York: John Wiley & Sons.

Siddiqi N (2006): Credit Risk Scorecards: developing and implementing intelligent credit scoring. New Jersey, Wiley.

Thomas LC, Edelman DB, Crook JN (2002): Credit Scoring and Its Applications. Philadelphia, SIAM Monographs on Mathematical Modeling and Computation.

See Also

Other model validation techniques: `blr_confusion_matrix()`, `blr_decile_capture_rate()`, `blr_decile_lift_chart()`, `blr_gains_table()`, `blr_gini_index()`, `blr_ks_chart()`, `blr_lorenz_curve()`, `blr_test_hosmer_lemeshow()`

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))
k <- blr_gains_table(model)
blr_roc_curve(k)
```

`blr_rsq_adj_count` *Adjusted count R2*

Description

Adjusted count r-squared.

Usage

```
blr_rsq_adj_count(model)
```

Arguments

`model`	An object of class `glm`.

Value

Adjusted count r-squared.

See Also

Other model fit statistics: blr_model_fit_stats(), blr_multi_model_fit_stats(), blr_pairs(), blr_rsq_cox_snell(), blr_rsq_effron(), blr_rsq_mcfadden_adj(), blr_rsq_mckelvey_zavoina(), blr_rsq_nagelkerke(), blr_test_lr()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))

blr_rsq_adj_count(model)
```

---

blr_rsq_count *Count R2*

Description

Count r-squared.

Usage

```
blr_rsq_count(model)
```

Arguments

model	An object of class glm.

Value

Count r-squared.

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))

blr_rsq_count(model)
```

blr_rsq_cox_snell *Cox Snell R2*

Description

Cox Snell pseudo r-squared.

Usage

```
blr_rsq_cox_snell(model)
```

Arguments

model	An object of class glm.

Value

Cox Snell pseudo r-squared.

References

Cox, D. R., & Snell, E. J. (1989). The analysis of binary data (2nd ed.). London: Chapman and Hall.

Maddala, G. S. (1983). Limited dependent and qualitative variables in economics. New York: Cambridge Press.

See Also

Other model fit statistics: blr_model_fit_stats(), blr_multi_model_fit_stats(), blr_pairs(), blr_rsq_adj_count(), blr_rsq_effron(), blr_rsq_mcfadden_adj(), blr_rsq_mckelvey_zavoina(), blr_rsq_nagelkerke(), blr_test_lr()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
            family = binomial(link = 'logit'))

blr_rsq_cox_snell(model)
```

blr_rsq_effron *Effron R2*

Description

Effron pseudo r-squared.

Usage

```
blr_rsq_effron(model)
```

Arguments

model An object of class glm.

Value

Effron pseudo r-squared.

References

Efron, B. (1978). Regression and ANOVA with zero-one data: Measures of residual variation. Journal of the American Statistical Association, 73, 113-121.

See Also

Other model fit statistics: blr_model_fit_stats(), blr_multi_model_fit_stats(), blr_pairs(), blr_rsq_adj_count(), blr_rsq_cox_snell(), blr_rsq_mcfadden_adj(), blr_rsq_mckelvey_zavoina(), blr_rsq_nagelkerke(), blr_test_lr()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))

blr_rsq_effron(model)
```

`blr_rsq_mcfadden` *McFadden's R2*

Description

McFadden's pseudo r-squared for the model.

Usage

```
blr_rsq_mcfadden(model)
```

Arguments

`model`	An object of class `glm`.

Value

McFadden's r-squared.

References

https://eml.berkeley.edu/reprints/mcfadden/zarembka.pdf

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))

blr_rsq_mcfadden(model)
```

`blr_rsq_mcfadden_adj` *McFadden's adjusted R2*

Description

McFadden's adjusted pseudo r-squared for the model.

Usage

```
blr_rsq_mcfadden_adj(model)
```

Arguments

`model`	An object of class `glm`.

Value

McFadden's adjusted r-squared.

References

https://eml.berkeley.edu/reprints/mcfadden/zarembka.pdf

See Also

Other model fit statistics: `blr_model_fit_stats()`, `blr_multi_model_fit_stats()`, `blr_pairs()`, `blr_rsq_adj_count()`, `blr_rsq_cox_snell()`, `blr_rsq_effron()`, `blr_rsq_mckelvey_zavoina()`, `blr_rsq_nagelkerke()`, `blr_test_lr()`

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
           family = binomial(link = 'logit'))

blr_rsq_mcfadden_adj(model)
```

---

blr_rsq_mckelvey_zavoina — *McKelvey Zavoina R2*

Description

McKelvey Zavoina pseudo r-squared.

Usage

```
blr_rsq_mckelvey_zavoina(model)
```

Arguments

`model`	An object of class `glm`.

Value

Cragg-Uhler (Nagelkerke) R2 pseudo r-squared.

References

McKelvey, R. D., & Zavoina, W. (1975). A statistical model for the analysis of ordinal level dependent variables. Journal of Mathematical Sociology, 4, 103-12.

See Also

Other model fit statistics: blr_model_fit_stats(), blr_multi_model_fit_stats(), blr_pairs(), blr_rsq_adj_count(), blr_rsq_cox_snell(), blr_rsq_effron(), blr_rsq_mcfadden_adj(), blr_rsq_nagelkerke(), blr_test_lr()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
           family = binomial(link = 'logit'))

blr_rsq_mckelvey_zavoina(model)
```

---

blr_rsq_nagelkerke *Cragg-Uhler (Nagelkerke) R2*

Description

Cragg-Uhler (Nagelkerke) R2 pseudo r-squared.

Usage

```
blr_rsq_nagelkerke(model)
```

Arguments

model	An object of class glm.

Value

Cragg-Uhler (Nagelkerke) R2 pseudo r-squared.

References

Cragg, S. G., & Uhler, R. (1970). The demand for automobiles. Canadian Journal of Economics, 3, 386-406.

Maddala, G. S. (1983). Limited dependent and qualitative variables in economics. New York: Cambridge Press.

Nagelkerke, N. (1991). A note on a general definition of the coefficient of determination.

See Also

Other model fit statistics: blr_model_fit_stats(), blr_multi_model_fit_stats(), blr_pairs(), blr_rsq_adj_count(), blr_rsq_cox_snell(), blr_rsq_effron(), blr_rsq_mcfadden_adj(), blr_rsq_mckelvey_zavoina(), blr_test_lr()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))

blr_rsq_nagelkerke(model)
```

`blr_segment` *Event rate*

Description

Event rate by segements/levels of a qualitative variable.

Usage

```
blr_segment(data, response, predictor)

## Default S3 method:
blr_segment(data, response, predictor)
```

Arguments

`data`	A `tibble` or `data.frame`.
`response`	Response variable; column in `data`.
`predictor`	Predictor variable; column in `data`.

Value

A tibble.

See Also

Other bivariate analysis procedures: `blr_bivariate_analysis()`, `blr_segment_dist()`, `blr_segment_twoway()`, `blr_woe_iv_stats()`, `blr_woe_iv()`

Examples

```
blr_segment(hsb2, honcomp, prog)
```

blr_segment_dist *Response distribution*

Description

Distribution of response variable by segements/levels of a qualitative variable.

Usage

```
blr_segment_dist(data, response, predictor)

## S3 method for class 'blr_segment_dist'
plot(
  x,
  title = NA,
  xaxis_title = "Levels",
  yaxis_title = "Sample Distribution",
  sec_yaxis_title = "1s Distribution",
  bar_color = "blue",
  line_color = "red",
  print_plot = TRUE,
  ...
)
```

Arguments

`data`	A `tibble` or a `data.frame`.
`response`	Response variable; column in `data`.
`predictor`	Predictor variable; column in `data`.
`x`	An object of class `blr_segment_dist`.
`title`	Plot title.
`xaxis_title`	X axis title.
`yaxis_title`	Y axis title.
`sec_yaxis_title`	Secondary y axis title.
`bar_color`	Bar color.
`line_color`	Line color.
`print_plot`	logical; if `TRUE`, prints the plot else returns a plot object.
`...`	Other inputs.

Value

A tibble.

See Also

Other bivariate analysis procedures: blr_bivariate_analysis(), blr_segment_twoway(), blr_segment(), blr_woe_iv_stats(), blr_woe_iv()

Examples

```
k <- blr_segment_dist(hsb2, honcomp, prog)
k

# plot
plot(k)
```

---

blr_segment_twoway   *Two way event rate*

Description

Event rate across two qualitative variables.

Usage

```
blr_segment_twoway(data, response, variable_1, variable_2)

## Default S3 method:
blr_segment_twoway(data, response, variable_1, variable_2)
```

Arguments

data	A tibble or data.frame.
response	Response variable; column in data.
variable_1	Column in data.
variable_2	Column in data.

Value

A tibble.

See Also

Other bivariate analysis procedures: blr_bivariate_analysis(), blr_segment_dist(), blr_segment(), blr_woe_iv_stats(), blr_woe_iv()

Examples

```
blr_segment_twoway(hsb2, honcomp, prog, female)
```

blr_step_aic_backward *Stepwise AIC backward elimination*

Description

Build regression model from a set of candidate predictor variables by removing predictors based on akaike information criterion, in a stepwise manner until there is no variable left to remove any more.

Usage

```
blr_step_aic_backward(model, ...)

## Default S3 method:
blr_step_aic_backward(model, progress = FALSE, details = FALSE, ...)

## S3 method for class 'blr_step_aic_backward'
plot(x, text_size = 3, print_plot = TRUE, ...)
```

Arguments

model	An object of class glm; the model should include all candidate predictor variables.
...	Other arguments.
progress	Logical; if TRUE, will display variable selection progress.
details	Logical; if TRUE, will print the regression result at each step.
x	An object of class blr_step_aic_backward.
text_size	size of the text in the plot.
print_plot	logical; if TRUE, prints the plot else returns a plot object.

Value

blr_step_aic_backward returns an object of class "blr_step_aic_backward". An object of class "blr_step_aic_backward" is a list containing the following components:

model	model with the least AIC; an object of class glm
candidates	candidate predictor variables
steps	total number of steps
predictors	variables removed from the model
aics	akaike information criteria
bics	bayesian information criteria
devs	deviances

References

Venables, W. N. and Ripley, B. D. (2002) Modern Applied Statistics with S. Fourth edition. Springer.

See Also

Other variable selection procedures: blr_step_aic_both(), blr_step_aic_forward(), blr_step_p_backward(), blr_step_p_forward()

Examples

```
## Not run:
model <- glm(honcomp ~ female + read + science + math + prog + socst,
data = hsb2, family = binomial(link = 'logit'))

# elimination summary
blr_step_aic_backward(model)

# print details of each step
blr_step_aic_backward(model, details = TRUE)

# plot
plot(blr_step_aic_backward(model))

# final model
k <- blr_step_aic_backward(model)
k$model


## End(Not run)
```

---

blr_step_aic_both *Stepwise AIC selection*

Description

Build regression model from a set of candidate predictor variables by entering and removing predictors based on akaike information criterion, in a stepwise manner until there is no variable left to enter or remove any more.

Usage

```
blr_step_aic_both(model, details = FALSE, ...)

## S3 method for class 'blr_step_aic_both'
plot(x, text_size = 3, ...)
```

Arguments

model	An object of class lm.
details	Logical; if TRUE, details of variable selection will be printed on screen.
...	Other arguments.
x	An object of class blr_step_aic_both.
text_size	size of the text in the plot.

Value

blr_step_aic_both returns an object of class "blr_step_aic_both". An object of class "blr_step_aic_both" is a list containing the following components:

model	model with the least AIC; an object of class glm
candidates	candidate predictor variables
predictors	variables added/removed from the model
method	addition/deletion
aics	akaike information criteria
bics	bayesian information criteria
devs	deviances
steps	total number of steps

References

Venables, W. N. and Ripley, B. D. (2002) Modern Applied Statistics with S. Fourth edition. Springer.

See Also

Other variable selection procedures: blr_step_aic_backward(), blr_step_aic_forward(), blr_step_p_backward(), blr_step_p_forward()

Examples

```
## Not run:
model <- glm(y ~ ., data = stepwise)

# selection summary
blr_step_aic_both(model)

# print details at each step
blr_step_aic_both(model, details = TRUE)

# plot
plot(blr_step_aic_both(model))

# final model
k <- blr_step_aic_both(model)
k$model
```

```
## End(Not run)
```

blr_step_aic_forward *Stepwise AIC forward selection*

Description

Build regression model from a set of candidate predictor variables by entering predictors based on chi square statistic, in a stepwise manner until there is no variable left to enter any more.

Usage

```
blr_step_aic_forward(model, ...)

## Default S3 method:
blr_step_aic_forward(model, progress = FALSE, details = FALSE, ...)

## S3 method for class 'blr_step_aic_forward'
plot(x, text_size = 3, print_plot = TRUE, ...)
```

Arguments

model	An object of class glm.
...	Other arguments.
progress	Logical; if TRUE, will display variable selection progress.
details	Logical; if TRUE, will print the regression result at each step.
x	An object of class blr_step_aic_forward.
text_size	size of the text in the plot.
print_plot	logical; if TRUE, prints the plot else returns a plot object.

Value

blr_step_aic_forward returns an object of class "blr_step_aic_forward". An object of class "blr_step_aic_forward" is a list containing the following components:

model	model with the least AIC; an object of class glm
candidates	candidate predictor variables
steps	total number of steps
predictors	variables entered into the model
aics	akaike information criteria
bics	bayesian information criteria
devs	deviances

References

Venables, W. N. and Ripley, B. D. (2002) Modern Applied Statistics with S. Fourth edition. Springer.

See Also

Other variable selection procedures: blr_step_aic_backward(), blr_step_aic_both(), blr_step_p_backward(), blr_step_p_forward()

Examples

```
## Not run:
model <- glm(honcomp ~ female + read + science, data = hsb2,
family = binomial(link = 'logit'))

# selection summary
blr_step_aic_forward(model)

# print details of each step
blr_step_aic_forward(model, details = TRUE)

# plot
plot(blr_step_aic_forward(model))

# final model
k <- blr_step_aic_forward(model)
k$model


## End(Not run)
```

---

blr_step_p_backward *Stepwise backward regression*

Description

Build regression model from a set of candidate predictor variables by removing predictors based on p values, in a stepwise manner until there is no variable left to remove any more.

Usage

```
blr_step_p_backward(model, ...)

## Default S3 method:
blr_step_p_backward(model, prem = 0.3, details = FALSE, ...)

## S3 method for class 'blr_step_p_backward'
plot(x, model = NA, print_plot = TRUE, ...)
```

Arguments

`model`	An object of class `lm`; the model should include all candidate predictor variables.
`...`	Other inputs.
`prem`	p value; variables with p more than `prem` will be removed from the model.
`details`	Logical; if `TRUE`, will print the regression result at each step.
`x`	An object of class `blr_step_p_backward`.
`print_plot`	logical; if `TRUE`, prints the plot else returns a plot object.

Value

`blr_step_p_backward` returns an object of class `"blr_step_p_backward"`. An object of class `"blr_step_p_backward"` is a list containing the following components:

`model`	model with the least AIC; an object of class `glm`
`steps`	total number of steps
`removed`	variables removed from the model
`aic`	akaike information criteria
`bic`	bayesian information criteria
`dev`	deviance
`indvar`	predictors

References

Chatterjee, Samprit and Hadi, Ali. Regression Analysis by Example. 5th ed. N.p.: John Wiley & Sons, 2012. Print.

See Also

Other variable selection procedures: `blr_step_aic_backward()`, `blr_step_aic_both()`, `blr_step_aic_forward()`, `blr_step_p_forward()`

Examples

```
## Not run:
# stepwise backward regression
model <- glm(honcomp ~ female + read + science + math + prog + socst,
  data = hsb2, family = binomial(link = 'logit'))
blr_step_p_backward(model)

# stepwise backward regression plot
model <- glm(honcomp ~ female + read + science + math + prog + socst,
  data = hsb2, family = binomial(link = 'logit'))
k <- blr_step_p_backward(model)
plot(k)

# final model
k$model
```

```
## End(Not run)
```

---

blr_step_p_both *Stepwise regression*

Description

Build regression model from a set of candidate predictor variables by entering and removing predictors based on p values, in a stepwise manner until there is no variable left to enter or remove any more.

Usage

```
blr_step_p_both(model, ...)

## Default S3 method:
blr_step_p_both(model, pent = 0.1, prem = 0.3, details = FALSE, ...)

## S3 method for class 'blr_step_p_both'
plot(x, model = NA, print_plot = TRUE, ...)
```

Arguments

model	An object of class lm; the model should include all candidate predictor variables.
...	Other arguments.
pent	p value; variables with p value less than pent will enter into the model.
prem	p value; variables with p more than prem will be removed from the model.
details	Logical; if TRUE, will print the regression result at each step.
x	An object of class blr_step_p_both.
print_plot	logical; if TRUE, prints the plot else returns a plot object.

Value

blr_step_p_both returns an object of class "blr_step_p_both". An object of class "blr_step_p_both" is a list containing the following components:

model	final model; an object of class glm
orders	candidate predictor variables according to the order by which they were added or removed from the model
method	addition/deletion
steps	total number of steps

predictors	variables retained in the model (after addition)
aic	akaike information criteria
bic	bayesian information criteria
dev	deviance
indvar	predictors

References

Chatterjee, Samprit and Hadi, Ali. Regression Analysis by Example. 5th ed. N.p.: John Wiley & Sons, 2012. Print.

Examples

```
## Not run:
# stepwise regression
model <- glm(y ~ ., data = stepwise)
blr_step_p_both(model)

# stepwise regression plot
model <- glm(y ~ ., data = stepwise)
k <- blr_step_p_both(model)
plot(k)

# final model
k$model


## End(Not run)
```

---

blr_step_p_forward *Stepwise forward regression*

Description

Build regression model from a set of candidate predictor variables by entering predictors based on p values, in a stepwise manner until there is no variable left to enter any more.

Usage

```
blr_step_p_forward(model, ...)

## Default S3 method:
blr_step_p_forward(model, penter = 0.3, details = FALSE, ...)

## S3 method for class 'blr_step_p_forward'
plot(x, model = NA, print_plot = TRUE, ...)
```

Arguments

model	An object of class lm; the model should include all candidate predictor variables.
...	Other arguments.
penter	p value; variables with p value less than penter will enter into the model
details	Logical; if TRUE, will print the regression result at each step.
x	An object of class blr_step_p_forward.
print_plot	logical; if TRUE, prints the plot else returns a plot object.

Value

blr_step_p_forward returns an object of class "blr_step_p_forward". An object of class "blr_step_p_forward" is a list containing the following components:

model	model with the least AIC; an object of class glm
steps	number of steps
predictors	variables added to the model
aic	akaike information criteria
bic	bayesian information criteria
dev	deviance
indvar	predictors

References

Chatterjee, Samprit and Hadi, Ali. Regression Analysis by Example. 5th ed. N.p.: John Wiley & Sons, 2012. Print.

Kutner, MH, Nachtscheim CJ, Neter J and Li W., 2004, Applied Linear Statistical Models (5th edition). Chicago, IL., McGraw Hill/Irwin.

See Also

Other variable selection procedures: blr_step_aic_backward(), blr_step_aic_both(), blr_step_aic_forward(), blr_step_p_backward()

Examples

```
## Not run:
# stepwise forward regression
model <- glm(honcomp ~ female + read + science, data = hsb2,
  family = binomial(link = 'logit'))
blr_step_p_forward(model)

# stepwise forward regression plot
model <- glm(honcomp ~ female + read + science, data = hsb2,
  family = binomial(link = 'logit'))
k <- blr_step_p_forward(model)
plot(k)
```

```
# final model
k$model


## End(Not run)
```

---

blr_test_hosmer_lemeshow   *Hosmer lemeshow test*

Description

Hosmer lemeshow goodness of fit test.

Usage

```
blr_test_hosmer_lemeshow(model, data = NULL)
```

Arguments

model	An object of class glm.
data	a tibble or data.frame.

References

Hosmer, D.W., Jr., & Lemeshow, S. (2000), Applied logistic regression(2nd ed.). New York: John Wiley & Sons.

See Also

Other model validation techniques: blr_confusion_matrix(), blr_decile_capture_rate(), blr_decile_lift_chart(), blr_gains_table(), blr_gini_index(), blr_ks_chart(), blr_lorenz_curve(), blr_roc_curve()

Examples

```
model <- glm(honcomp ~ female + read + science, data = hsb2,
             family = binomial(link = 'logit'))

blr_test_hosmer_lemeshow(model)
```

blr_test_lr *Likelihood ratio test*

Description

Performs the likelihood ratio test for full and reduced model.

Usage

```
blr_test_lr(full_model, reduced_model)

## Default S3 method:
blr_test_lr(full_model, reduced_model)
```

Arguments

full_model	An object of class glm; model with all predictors.
reduced_model	An object of class glm; nested model. Optional if you are comparing the full_model with an intercept only model.

Value

Two tibbles with model information and test results.

See Also

Other model fit statistics: blr_model_fit_stats(), blr_multi_model_fit_stats(), blr_pairs(), blr_rsq_adj_count(), blr_rsq_cox_snell(), blr_rsq_effron(), blr_rsq_mcfadden_adj(), blr_rsq_mckelvey_zavoina(), blr_rsq_nagelkerke()

Examples

```
# compare full model with intercept only model
# full model
model_1 <- glm(honcomp ~ female + read + science, data = hsb2,
            family = binomial(link = 'logit'))

blr_test_lr(model_1)

# compare full model with nested model
# nested model
model_2 <- glm(honcomp ~ female + read, data = hsb2,
            family = binomial(link = 'logit'))

blr_test_lr(model_1, model_2)
```

blr_woe_iv *WoE & IV*

Description

Weight of evidence and information value. Currently avialable for categorical predictors only.

Usage

```
blr_woe_iv(data, predictor, response, digits = 4, ...)

## S3 method for class 'blr_woe_iv'
plot(
  x,
  title = NA,
  xaxis_title = "Levels",
  yaxis_title = "WoE",
  bar_color = "blue",
  line_color = "red",
  print_plot = TRUE,
  ...
)
```

Arguments

data	A tibble or data.frame.
predictor	Predictor variable; column in data.
response	Response variable; column in data.
digits	Number of decimal digits to round off.
...	Other inputs.
x	An object of class blr_segment_dist.
title	Plot title.
xaxis_title	X axis title.
yaxis_title	Y axis title.
bar_color	Color of the bar.
line_color	Color of the horizontal line.
print_plot	logical; if TRUE, prints the plot else returns a plot object.

Value

A tibble.

References

Siddiqi N (2006): Credit Risk Scorecards: developing and implementing intelligent credit scoring. New Jersey, Wiley.

See Also

Other bivariate analysis procedures: blr_bivariate_analysis(), blr_segment_dist(), blr_segment_twoway(), blr_segment(), blr_woe_iv_stats()

Examples

```
# woe and iv
k <- blr_woe_iv(hsb2, female, honcomp)
k

# plot woe
plot(k)
```

---

blr_woe_iv_stats *Multi variable WOE & IV*

Description

Prints weight of evidence and information value for multiple variables. Currently avialable for categorical predictors only.

Usage

```
blr_woe_iv_stats(data, response, ...)
```

Arguments

data	A data.frame or tibble.
response	Response variable; column in data.
...	Predictor variables; column in data.

See Also

Other bivariate analysis procedures: blr_bivariate_analysis(), blr_segment_dist(), blr_segment_twoway(), blr_segment(), blr_woe_iv()

Examples

```
blr_woe_iv_stats(hsb2, honcomp, prog, race, female, schtyp)
```

hsb2 *High School and Beyond Data Set*

Description

A dataset containing demographic information and standardized test scores of high school students.

Usage

```
hsb2
```

Format

A data frame with 200 rows and 11 variables:

id id of the student
female gender of the student
race ethnic background of the student
ses socio-economic status of the student
schtyp school type
prog program type
read scores from test of reading
write scores from test of writing
math scores from test of math
science scores from test of science
socst scores from test of social studies
honcomp 1 if write > 60, else 0

Source

http://www.ats.ucla.edu/stat/spss/whatstat/whatstat.htm

stepwise *Dummy Data Set*

Description

Dummy Data Set

Usage

```
stepwise
```

Format

An object of class `data.frame` with 20000 rows and 7 columns.

Index